Expert Systems With Applications 66 (2016) 234–244

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

A collaborative filtering method for music recommendation using playing coefficients for artists and users

Diego Sánchez-Moreno, Ana B. Gil González, M. Dolores Muñoz Vicente, Vivian F. López Batista, María N. Moreno García*

Department of Computing and Automation, University of Salamanca, Salamanca, Spain

ARTICLE INFO

Article history:
Received 22 June 2016
Revised 10 September 2016
Accepted 11 September 2016
Available online 13 September 2016

Keywords:
Collaborative filtering
Music recommendation
Data mining
Sparsity
Gray-sheep

ABSTRACT

The great quantity of music content available online has increased interest in music recommender systems. However, some important problems must be addressed in order to give reliable recommendations. Many approaches have been proposed to deal with cold-start and first-rater drawbacks; however, the problem of generating recommendations for gray-sheep users has been less studied. Most of the methods that address this problem are content-based, hence they require item information that is not always available. Another significant drawback is the difficulty in obtaining explicit feedback from users, necessary for inducing recommendation models, which causes the well-known sparsity problem. In this work, a recommendation method based on playing coefficients is proposed for addressing the above-mentioned shortcomings of recommender systems when little information is available. The results prove that this proposal outperforms other collaborative filtering methods, including those that make use of user attributes.

1. Introduction

Current music platforms make available to users a large number of songs through web sites and mobile apps. These abilities have led to the extension of the overload problem, which has its origin in the context of information retrieval, to this kind of applications, since users have difficulties in finding out the music they like. In order to make this task easier, many of the platforms have searching services and some of them are endowed with recommendation mechanisms. However, in the last case, it is necessary to address some of the usual problems of recommender systems.

Collaborative filtering (CF) methods are widely used in recommender systems. They provide recommendations based on ratings that users give to items. The results of these techniques are quite good; however, the difficulty in obtaining explicit feedback in the form of ratings from the users causes the sparsity problem, which takes place when the number of available ratings for the items to be recommended is small. This is the main drawback that prevents the application of this approach in many systems. A way to address this problem is to derive implicit ratings from user behavior.

There are two approaches for collaborative filtering: memory-based and model-based algorithms. Memory-based algorithms, also known as nearest-neighbor methods, were the earliest used. They treat all user items in order to find users with similar preferences (neighbors). The advantage of these algorithms is the quick incorporation of the most recent information, but they can present scalability problems given that the search for neighbors in large databases is slow. Model-based CF algorithms deal with scalability problems by using the ratings from users for computing item similarity instead of user similarity, considering that items are similar if they are liked/disliked by the same users. Thus, recommendations can be provided to users, given that users are expected to have similar preferences for similar items.

Additional drawbacks presented by CF methods are early-rater (first-rater) and cold-start problems. The first one takes place when new products are introduced in the system. These items have never been rated, therefore they cannot be recommended. Cold-start problem affect new users, who cannot receive recommendations since they have no evaluations about products.

Content-based algorithms have been proposed as an alternative to CF methods in order to deal with the shortcomings discussed previously. These methods can be used for recommending any kind of item by making use of its features. Thus, new items can be recommended according to their similarity to other items for which the user showed interest in the past. Gray-sheep users also generate a problem in recommender systems that has sometimes been addressed with content-based methods. These are users who have unusual preferences; thus they do not have enough neighbors for computing reliable recommendations.

* Corresponding author.
E-mail addresses: sanchezhh@gmail.com (D. Sánchez-Moreno), abg@usal.es (A.B. Gil González), mariado@usal.es (M.D. Muñoz Vicente), vivian@usal.es (V.F. López Batista), mmg@usal.es (M.N. Moreno García).

http://dx.doi.org/10.1016/j.eswa.2016.09.019
0957-4174/

Most of the current recommender systems use hybrid techniques aiming at taking advantage of the strengths of both approaches and avoiding their drawbacks. These methods take into account the preferences of other users as well as the characteristics of items and users (age, gender, occupation…). Therefore, new users can receive recommendations depending on their characteristics.

Although there are many proposals in the literature for dealing with the weaknesses of recommender systems, the gray-sheep problem has received less attention, and is mainly addressed by means of hybrid approaches that involve some content-based technique. Their results are usually good but these methods require information about items and users that often is not available. In the context of music recommender systems, content-based filtering algorithms use musical content for inducing the models; therefore, a complex extraction task of music features is necessary. In this work, the proposed recommendation methodology addresses the above-mentioned drawback when little information is available. The recommendation process could be incorporated into any music platform as long as it stores user and artist identification and the number of times the user plays a song in the platform, without the need for collecting rating data.

The rest of the paper is organized as follows: Section 2 includes a description of the state of the art of recommendation methods with special focus on collaborative filtering. The proposed methodology is described in Section 3 and the empirical study conducted for its validation is reported in Section 4. Finally, the conclusions and future work are given in Section 5.

2. Related work

Most of the current recommender systems use some CF based approach. The aim of CF is to predict the rating that a target user would give to an item taking into account users having similar preferences to this target user regarding previously rated items. The GroupLens research system for Usenet news (Resnick, Iacovou, Suchack, Bergstrom, & Riedl, 1994) was the first recommender system using CF, and Ringo (Sarwar, Karypis, Konstan, & Riedl, 2001) was one of the first and most popular music recommender systems based on CF.

CF requires user-explicit expression of personal preferences for items in the form of ratings, which are usually difficult to obtain. This fact is at the root of one of the main drawbacks of this approach, the sparsity problem, which arises when the number of ratings needed for prediction is greater than the number of the ratings obtained from the users. The time that users spend examining the items is an alternative way to obtain implicit user preferences (Sarwar et al., 2001) but it requires processing of log files and this implicit information about user preferences is not as reliable as the explicit ratings. In the music recommender area several ways of dealing with this problem have been proposed. The access history of users is taken as an implicit way of obtaining user interests in a music recommendation system based on music and user grouping (Chen & Chen, 2005). In several works where the last.fm database is used, the times that the users play the songs (play counts) are converted to ratings by means of different functions (Lee & Lee, 2015; Vargas & Castells, 2011). The ratings, whether implicit or explicit, are arranged in a user-items rating matrix. Empty elements of the matrix represent items not rated by the corresponding users.

In memory-based (user-based or user-user) CF methods (Resnick et al., 1994) the predictions for a given user, called the active user, are based on that person's nearest neighbors. Neighbors are users who have similar preferences to the active user since they have rated items in common with a similar score. These methods need to use the entire rating matrix to compute the similarity between users. In consequence, the computation time grows linearly with both the number of customers and the number of items in the system. This drop in performance, known as the scalability problem, has a direct impact on the user response time since similarity is computed at recommendation time. There are different measures for obtaining the similarity; however, the most extended ones are the Pearson correlation coefficient and cosine similarity (Breese, Heckerman, & Kadie, 1998). The Pearson correlation usually provides better results than cosine similarity but its computational cost is higher.

Model-based (item-based or item-item) CF was proposed in Sarwar et al. (2001) with the aim of avoiding the scalability problem, has a direct impact on the user response time since similarity is computed at recommender time the scalability problems (Schafer, Konstant, & Riedl, 2001) associated with memory-based methods by precomputing the similarities between items. This can be done since it is expected that new ratings given to items in large rating databases do not significantly change between-item similarity, especially for much-rated items. On the contrary, precomputing the similarities between users would not be effective given that the neighborhood of a user is obtained from both his ratings and the ratings of other users, which undergo continuous changes and additions (Ekstrand, Riedl, & Konstan, 2010). There are several procedures for computing item similarity; however, cosine similarity is the most extended one because of its simplicity, efficiency and better results regarding accuracy than the Pearson coefficient. Recommendations provided by item-based methods usually have less quality than those provided by user-based approaches, but they can be suitable to be applied in large-scale systems where scalability is a serious problem. For example, they have been used in popular systems like Amazon (Lucas et al., 2013).

Another kind of model-based algorithms builds a predictive model of user preferences by means of data mining techniques. Besides the ratings, other attributes of items and/or users can be used for inducing the model, so this process also involves a content-based approach. Data mining methods usually behave better against sparsity, especially association-based methods (Lucas, Laurent, Moreno, & Teisseire, 2012). Moreover, scalability problems are avoided since predictive models are already built when the user requests recommendations; thus, the building time has no impact on the user response time. The main inconvenience of these techniques is the need for frequent updating of the models in order to incorporate the most recent information generated by users. Moreover, data mining methods require more information than the simple rating data.

Common shortcomings of both user-based and item-based CF are the cold-start and early-rater (first-rater) problems, described in the previous section. In these cases, when recommendations cannot be provided to new users or new products cannot be recommended, respectively, then content-based methods can be applied. They were first used to recommend text documents by comparing their contents and the contents of other documents associated with the user profile but without taking into account the opinion of other users (Lee, Kim, & Rhee, 2001). Currently, they have been extended to other domains by replacing document contents by other characteristics of the items (Billsus & Pazzani, 1999; Krulwich & Burkey, 1996). They also take advantage of the similarity between items but they do not need rating data since they make use of other features of the items for computing the similarity. Some content-based approaches use distance metrics such as cosine similarity but others resort to data mining methods. In the music field, metadata of the items, such as title, artist, genre and lyrics, can be exploited as content attributes, but also audio features like timbre, melody, rhythm or harmony. In Tzanetakis (2002), similarity was determined from chord structure (spectrum, rhythm and harmony). Melody style is the music feature used in Kuo and Shan (2002) for music recommendation. A